3-Heights™ PDF Optimization Service

Version 4.5

User Manual

Contact: pdfsupport@pdf-tools.com

Owner: **PDF Tools AG**

Kasernenstrasse 1
8184 Bachenbülach
Switzerland

http://www.pdf-tools.com

Copyright © 2000-2015

Table of Contents

1 Introduction 5
1.1 Description 5
1.2 Functions 5
Features 6
Formats 6
Compliance 7
1.3 Service 7
1.4 Operating Systems 7
2 Installation 8
2.1 Overview 8
2.2 Installing the Service 8
2.3 Editing the Configuration File "pdfoptimizesvr.ini" 9
Retrieve Information about Available Options 9
2.4 Creating, Starting and Running the Service 10
2.5 Error Log Files 12
2.6 State Diagram of the Service 13
3 License Management 14
3.1 Graphical License Manager Tool 14
List all installed license keys 14
Add and delete license keys 14
Display the properties of a license 14
Select between different license keys for a single product 15
3.2 Platform considerations 15
Windows 15
4 Optimization Process 15
4.1 Images 15
Relevant Factors for the File Size 15
Examples 16
Optimizing Raster Images 16
Mixed Raster Content (MRC) Optimization for Images 17
4.2 Optimize Fonts 19
4.3 Suggested Settings for the Web 19
4.4 Suggested Settings for Printing 20
5 Reference Manual 21
5.1 Service Control Commands 21
-c Create Service 21
-s Start Service 21
-a Pause Service 21
-o Continue Service 21
-t Stop Service 21
-d Delete Service 22
-q Query Current Status of Service 22
-i List Options for the Configuration File 22
-x Run as Executable 22
5.2 File and Folder Options 23

© PDF Tools AG – Premium PDF Technology

Configuration File *PdfOptimizeSvr.ini* 23
-lk Set License Key 23
-w Set the Watched Folder 24
-wd Set the Drop-In Folder 24
-wfs Process only files with certain extensions 24
-wfi Ignore files with certain extensions 24
`-wfi .temp.tmp` 24
-p Set the Permission Flags 24
-pw Read an Encrypted PDF File 25
-o Set the Owner Password 25
-u Set the User Password 25
5.3 Optimization Options 26
Compression Values for Images 26
-c Set the Color Conversion 27
-cff Compress Type1 fonts (convert to CFF) 28
-cms Set the color management engine 28
Set the Resolution and Threshold per Image Type 28
-dr Set the Resolution in DPI 29
-dt Set the Threshold in DPI 29
-fb Set the Compression Types for Bi-tonal Images 29
-fc Set the Compression Types for Color and Grayscale Images 30
-ff Force Compression Conversion 31
-fi Set the Compression Types for Indexed (Paletted) Images 31
-fv Set the Minimum PDF Version 32
-id Set Value in the Document Information Dictionary 32
-isc Enable Conservative Image Processing Strategy 32
-m Merge Embedded Font Programs 32
-ml Set Compression Type for MRC Layers 33
-mlr Set the Resolution in DPI for MRC Layers 33
-mlq Set the Image Quality for MRC Layers 33
-mm Set Compression Type for the MRC Mask 33
-mp Set Compression Type for MRC Cut-Out Pictures 33
-o Set the Owner Password 33
-oc Clip Images 34
-od Optimize Resources 34
-ol Linearize Only 34
-or Remove Redundant Objects 34
-ow Linearize the Output File 34
-p Set the Permission Flags 34
-pw Read an Encrypted PDF File 35
-q Set the Compression Quality 35
-rs Remove Embedded Standard Fonts 36
-s Subset Fonts 36
Strip the File 36
-u Set User Password 37
5.4 Service Settings 37
Autodelete Successfully Processed Files 37
Job Number Prefix 37
Logpath 38
Polling Interval 38
6 Tips, Tricks and Troubleshooting 39

© PDF Tools AG – Premium PDF Technology

6.1 The Output File Is Still Too Large 39
6.2 The Output File Is Larger Than the Input File 39
6.3 The Selected Compression Type is Not Applied 40
6.4 The Output Document Is Not Encrypted 40

© PDF Tools AG – Premium PDF Technology

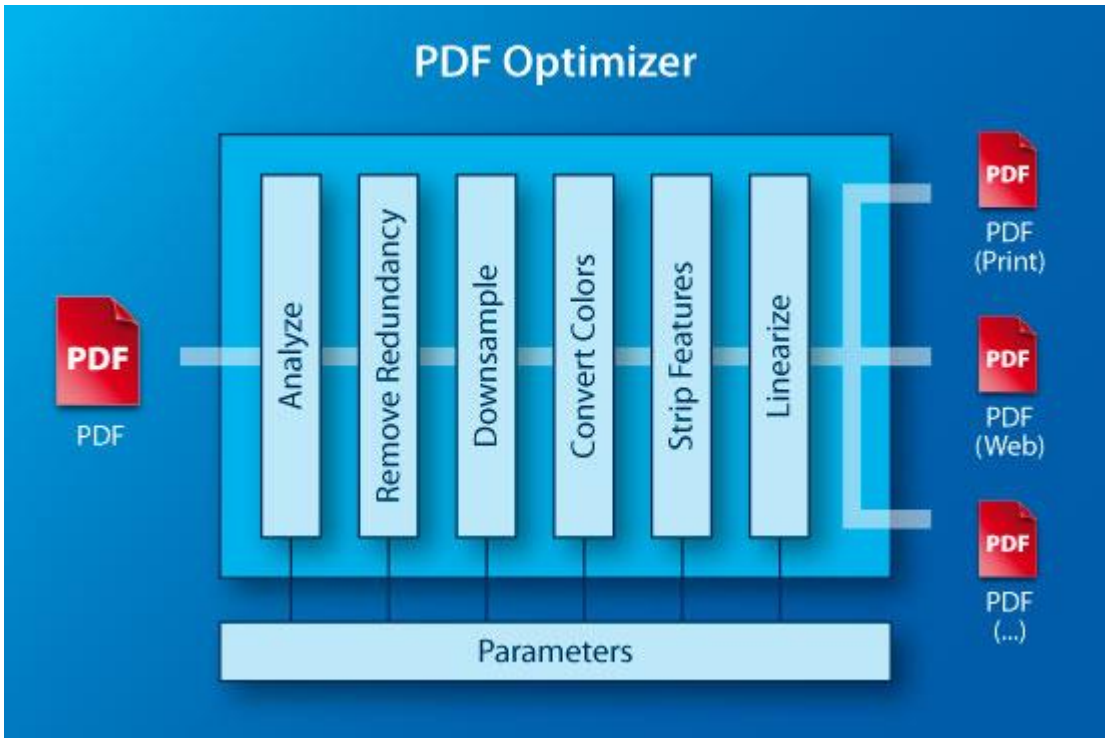
1 Introduction

1.1 Description

The 3-Heights™ PDF Optimization Service optimizes PDF files to enable their use as high resolution files for printing or, with less resolution, for electronic document exchange or space-saving document archiving.

Many processes produce very large PDF files that are not suitable for electronic document exchange. Users are then tempted to convert the PDF documents into other formats, but this only makes the situation even worse. The correct approach, and the easiest, is to optimize large PDF documents.

This process optimizes fonts and images to the best possible size and quality. It also removes redundant document content and linearizes PDF documents to enable fast web display.



1.2 Functions

The use of the latest compression algorithms enables the tool to reduce the memory space requirements for images or lessen their resolution, remove redundant and alternative information, optimize fonts through summarization or subsetting, convert colors and linearize the PDF.

© PDF Tools AG – Premium PDF Technology

Features

- Optimization for Electronic Document Exchange, Web Publishing and Archiving
- Customized compression of bi-tonal, monochrome and color images
- Define image resolution in dots per inch
- Define threshold value for down-sampling
- Set the quality index of lossy compression
- Automatically select best compression type for images
- Perform mixed raster content (MRC) optimization for images
- Remove invisible parts of images
- Linearization (fast web display)
- Compile and subset fonts
- Read encrypted input files
- Encrypt and set access authorization for the output file
- Process memory-resident files
- Removal of:
    - Redundant objects
    - Obsolete objects stemming from previous changes to the file
    - Embedded standard fonts (e.g. Courier, Arial, Times)
    - Embedded, non-symbolic fonts
    - Unnecessary file information
    - Article threads
    - Alternative images
    - Metadata
    - Page piece information
    - Document structure tree including markup
    - Miniature page preview images
    - Spider (web capture) information
- Remove or clear form fields and annotations

Optimize for Printing:

- Color conversion (to RGB, CMYK or grayscale)
- Allow high print quality
- Set minimum PDF version of the output file

Formats

Input Formats:

© PDF Tools AG – Premium PDF Technology

- PDF 1.x (e.g. PDF 1.4, PDF 1.5.)

Target Formats:

- PDF 1.x (e.g. PDF 1.4, PDF 1.5)

Compliance

Standards: ISO 32000 (PDF 1.7)

1.3 Service

The 3-Heights™ PDF Optimizer Service is a ready-to-use Windows Service that allows to install a Windows NT service process to automatically optimize PDF documents from watched folders by optimizing fonts, images and redundant information.

The PDF Optimization Service combines three programs in one executable.

1. An optimization service, that can be run on a Windows platforms (2000 or later). The service can be created, started, paused, resumed, stopped and deleted via the Windows service control panel and reports to the application log of the Windows event log panel.
2. A command line interface to control the PDF optimization service. By means of this interface the service can be created, started, stopped and deleted, etc.

- A query program which can be used to retrieve information about available options.

1.4 Operating Systems

- Windows XP, Vista, 7, 8, 8.1 – 32 and 64 bit
- Windows Server 2003, 2008, 2008-R2, 2012, 2012-R2 – 32 and 64 bit

© PDF Tools AG – Premium PDF Technology

2 Installation

2.1 Overview

The PDF Optimization Service is configured by the file *pdfoptimizesvr.ini*, which needs to be located in the same directory as the executable *pdfoptimizesvr.exe*. Before starting the service, the configuration file needs to be adjusted. How this is done is described in the chapter *Editing the Configuration File "pdfoptimizesvr.ini"*.

Once configured, the service can be created, started, paused, continued, stopped and deleted via the command line. To use the create and delete functions, administrator permissions are required. To start and stop the service, operator permissions are required.

When the service is running, it processes PDF documents that are copied or moved into watched folders.

They are then renamed and moved to the folder Jobs. The renaming gives the PDF a 16 character long timestamp to create unique job tickets.

2.2 Installing the Service

The retail version of the 3-Heights™ PDF Optimization Service comes as a ZIP archive containing various files including runtime binary executable code, documentation and license terms.

1. Download the ZIP archive of the product from your download account at www.pdf-tools.com.
2. Open the ZIP archive.
3. Check the appropriate option to preserve file paths (folder names) and unzip the archive to a local folder (e.g. *C:\program files\pdf-tools*).
4. The unzip process now creates the following subdirectories:
    - Bin: Contains the runtime executable binary code
    - Doc: Contains documentation files

© PDF Tools AG – Premium PDF Technology

2.3 Editing the Configuration File "pdfoptimizesvr.ini"

Before starting the PDF Optimization Service for the first time, the file *pdfoptimizesvr.ini* needs to be modified. Editing this file while the service is running has no impact. The service first needs to be stopped and restarted after the modification. When opening this file with a text editor, it looks like this:

```
[PdfOptimizeSvr]

AutoDelete=True

Threads=2

Thread1=-w C:\WebOptimize -c 1 -dt 120 -dr 96 -q 75 -od -or -ow -ss -st

Thread2=-w C:\PrintOptimize -dt 1500 -dr 1200 -od -or
```

The meaning is as following:

AutoDelete=True This option automatically deletes a PDF file after it is processed. When set to False, successfully processed files will be copied to the subfolder *Succeeded*\.

Threads=2 stands for the total amount of concurrent threads. Each thread can have its own assigned settings. Usually 1 thread corresponds to 1 watched folder, this is not required however.

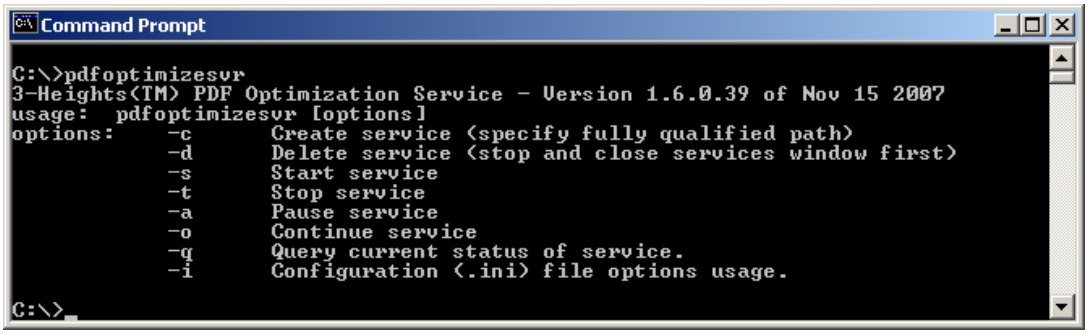
Thread1= sets the options such as name of watched folder and settings for Thread1.

-w C:\WebOptimize sets Thread1 to use this directory as watched folder. If the directory has blanks in its path, the entire path needs to be set in quotation marks (e.g. –w "C:\program files\optimize for web")

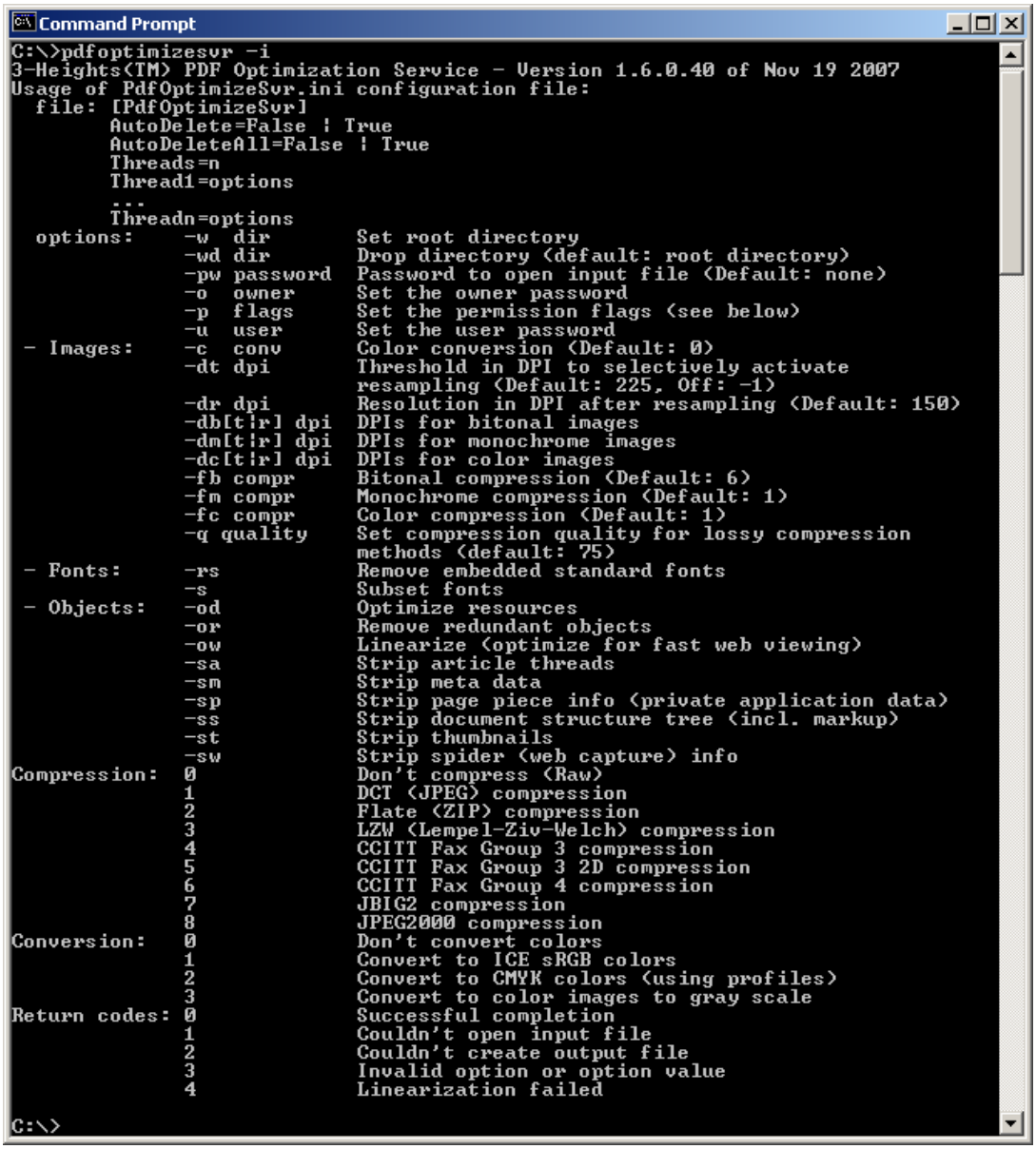
-c 1 –dt 120… sets the options for this watched folder.

Retrieve Information about Available Options

Open a shell and type PdfOptimizeSvr without parameters. This lists the usage, displaying all available service related options, as shown in the image below.


Command Prompt
C:\>pdfoptimizesvr
3-Heights(TM) PDF Optimization Service - Version 1.6.0.39 of Nov 15 2007
usage: pdfoptimizesvr [options]
options: -c Create service (specify fully qualified path)
-d Delete service (stop and close services window first)
-s Start service
-t Stop service
-a Pause service
-o Continue service
-q Query current status of service.
-i Configuration (.ini) file options usage.
C:\>

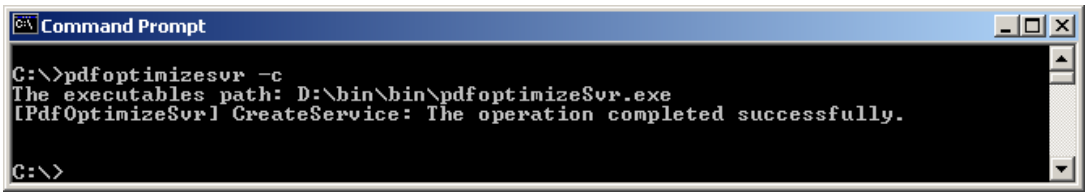
Use the option –i to get a list of all available configuration options:


Command Prompt
C:\>pdfoptimizesvr -i
3-Heights(TM) PDF Optimization Service - Version 1.6.0.40 of Nov 19 2007
Usage of PdfOptimizeSvr.ini configuration file:
file: [PdfOptimizeSvr]
AutoDelete=False | True
AutoDeleteAll=False | True
Threads=n
Thread1=options
...
Threadn=options
options: -w dir Set root directory
-wd dir Drop directory (default: root directory)
-pw password Password to open input file (Default: none)
-o owner Set the owner password
-p flags Set the permission flags (see below)
-u user Set the user password
- Images: -c conv Color conversion (Default: 0)
-dt dpi Threshold in DPI to selectively activate resampling (Default: 225, Off: -1)
-dr dpi Resolution in DPI after resampling (Default: 150)
-db[t|r] dpi DPIs for bitonal images
-dm[t|r] dpi DPIs for monochrome images
-dc[t|r] dpi DPIs for color images
-fb compr Bitonal compression (Default: 6)
-fm compr Monochrome compression (Default: 1)
-fc compr Color compression (Default: 1)
-q quality Set compression quality for lossy compression methods (default: 75)
- Fonts: -rs Remove embedded standard fonts
-s Subset fonts
- Objects: -od Optimize resources
-or Remove redundant objects
-ow Linearize (optimize for fast web viewing)
-sa Strip article threads
-sm Strip meta data
-sp Strip page piece info (private application data)
-ss Strip document structure tree (incl. markup)
-st Strip thumbnails
-sw Strip spider (web capture) info
Compression: 0 Don't compress (Raw)
1 DCT (JPEG) compression
2 Flate (ZIP) compression
3 LZW (Lempel-Ziv-Welch) compression
4 CCITT Fax Group 3 compression
5 CCITT Fax Group 3 2D compression
6 CCITT Fax Group 4 compression
7 JBIG2 compression
8 JPEG2000 compression
Conversion: 0 Don't convert colors
1 Convert to ICE sRGB colors
2 Convert to CMYK colors (using profiles)
3 Convert to color images to gray scale
Return codes: 0 Successful completion
1 Couldn't open input file
2 Couldn't create output file
3 Invalid option or option value
4 Linearization failed
C:\>

2.4 Creating, Starting and Running the Service

Once the configuration file is completed, the PDF Optimization Service can be controlled via the command line. To create or delete the service, administrator permissions are required.

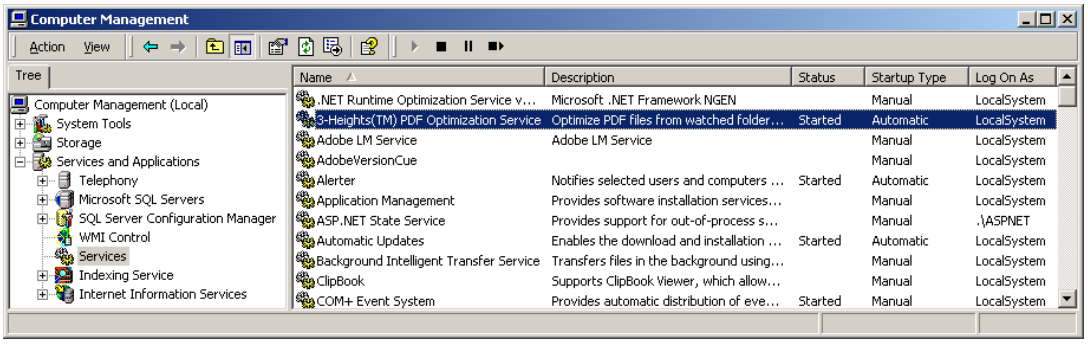
1. To create the service, use the option –c. This function will automatically search for the executable path.



Important: It is essential that *PdfOptimizerSvr.exe* be on a **non-mapped** drive.

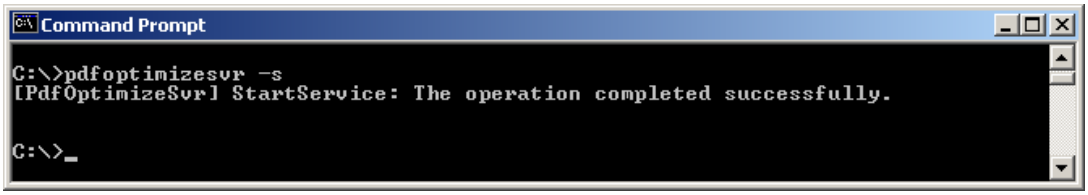
After executing this command, the service is created. It is now visible in the "Computer Management" window under "Services". To open the "Computer Management" window, go to Start -> Control Panel -> Administrative Tools -> Computer Management, or simply right-click the icon "My Computer" on the desktop and select "manage".

When the service is created, it will appear as *3-Heights(TM)PDF Optimization Service* as shown in the image below.

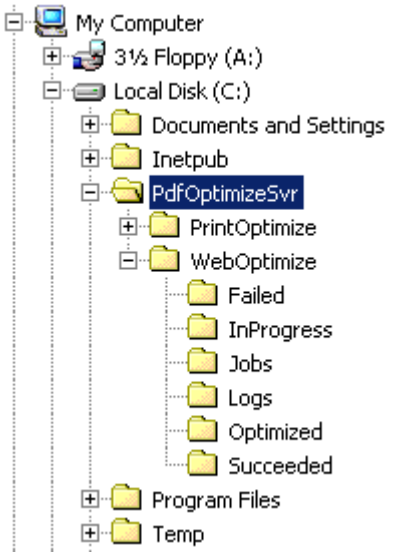

Computer Management
Action View
Tree
Computer Management (Local)
System Tools
Storage
Services and Applications
Telephony
Microsoft SQL Servers
SQL Server Configuration Manager
WMI Control
Services
Indexing Service
Internet Information Services
Name
Description
Status
Startup Type
Log On As
.NET Runtime Optimization Service v... Microsoft .NET Framework NGEN Manual LocalSystem
3-Heights(TM) PDF Optimization Service Optimize PDF files from watched folder... Started Automatic LocalSystem
Adobe LM Service Adobe LM Service Manual LocalSystem
AdobeVersionCue Manual LocalSystem
Alerter Notifies selected users and computers ... Started Automatic LocalSystem
Application Management Provides software installation services... Manual LocalSystem
ASP.NET State Service Provides support for out-of-process s... Manual .\ASPNET
Automatic Updates Enables the download and installation ... Started Automatic LocalSystem
Background Intelligent Transfer Service Transfers files in the background using... Manual LocalSystem
ClipBook Supports ClipBook Viewer, which allow... Manual LocalSystem
COM+ Event System Provides automatic distribution of eve... Started Manual LocalSystem

By default, the user is set to LocalSystem. After the service is created, the user can be changed. This will be required in a situation where a network share is used as a watched folder and the process needs to run under a user with the appropriate permission rights since the account LocalSystem does not have any permissions on remote systems.

2. After it is created and the logon settings are adjusted as required, the service can be started with the option –s. The path can be omitted if the *pdfoptimizesvr.exe* is included in the environment variable "Path".



Once the 3-Heights™ PDF Optimization Service is started, a series of watched folders and sub folders, corresponding to the configuration file *pdfoptimize.ini*, is created.



- PDF documents can now be copied or drag-and-dropped into the watched folder (*PrintOptimize*\ and *WebOptimize*\).
- From there it will be added a 16 byte time stamp prefix and moved to the folder *Jobs*\. A file that is currently being processed is moved from *Jobs*\ to *InProgress*\.
- Successfully optimized output files are available in the folder *Optimized*\. If "AutoDelete" is set to False in the configuration file, the successfully processed input file is copied to the folder *Succeeded*\, if set to True, it is deleted.
- Documents that cannot be processed are always moved to the folder *Failed*\. An error log text file is created for failed documents. This log has the same name as the input file, but the extension ".txt". The log file is available in the sub folder *Logs*\. Additionally a entry in the log file of the thread is created.

3. To stop the service, use the option –t. To restart use –s again.
4. To delete the service, stop it first, then use the option –d to delete.

2.5 Error Log Files

There are two types of error log files.

1. The log file per thread. Each thread (watched folder) has a log file. The log file is locked by the service as long as the service is running.
   - The log file contains general messages such as:

     ```
     -Service thread for directory c:\PdfOptimizeSvr started.

     -Thread for directory c:\PdfOptimizeSvr terminated.
     ```

   - Error messages such as:

     ```
     * Error 0 while opening file c:\PdfOptimizeSvr\Jobs\Job-
     01C5EE97-6F2667EC_document.pdf.
     ```

2. The for each document which could not be processed, there is an error log file created. The log file is written to the folder *Log*\ and has the same file name as

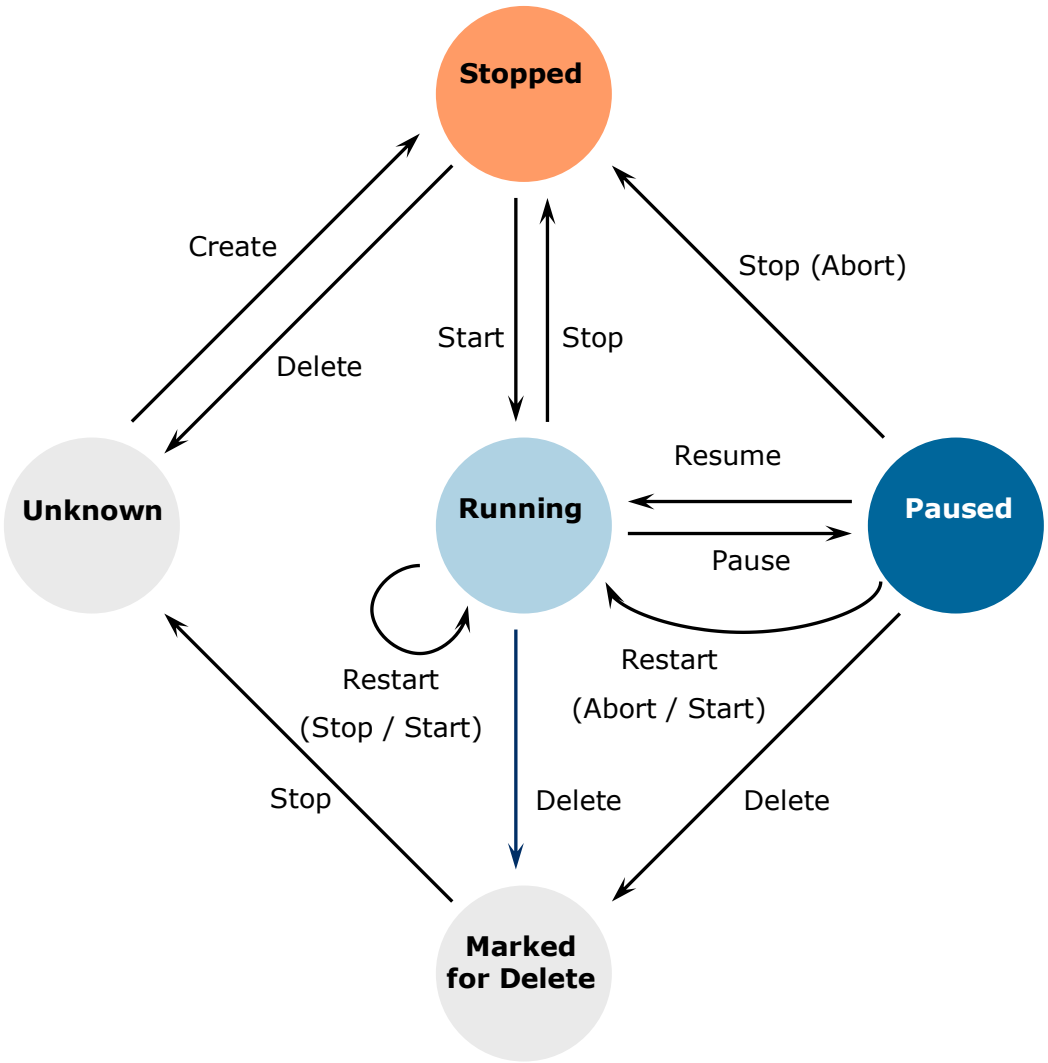
the input document, with extension ".txt". There is no error log available if a document cannot be opened (e.g. is not a PDF document).

2.6 State Diagram of the Service

The 3-Heights PDF Optimization Service behaves as described in the state diagram below:

If "Stop" if called when the service is in the state "Paused", the current job is aborted. This means the current page is finished processing, then the job is terminated.

If "Stop" is called when the service is the state "Running", the current job (all pages) is finished. Then the service is stopped.

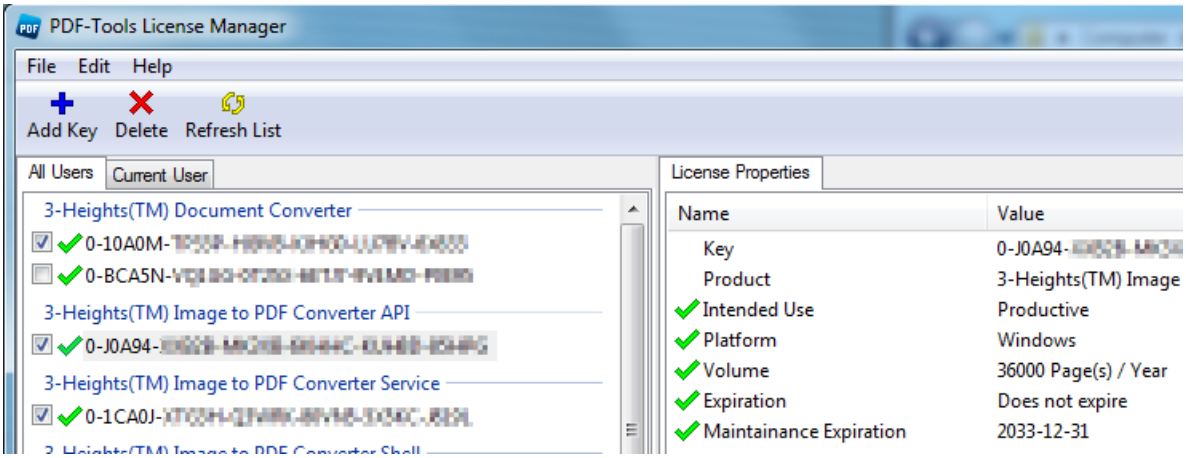


3 License Management

For services, there is only one possibility to pass the license key to the application: The license key is installed using the GUI tool (Graphical user interface). This is the easiest way if the licenses are managed manually.

3.1 Graphical License Manager Tool

The GUI tool *LicenseManager.exe* is located in the *bin* directory of the product kit.



List all installed license keys

The license manager always shows a list of all installed license keys in the left pane of the window. This includes licenses of other PDF Tools products.

The user can choose among:

- Licenses available for all users. Administrator rights are needed for modifications.
- Licenses available for the current user only.

Add and delete license keys

License keys can be added or deleted with the "Add Key" and "Delete" buttons in the toolbar.

- The "Add key" button installs the license key into the currently selected list.
    - Note: Services run by default under the LOCAL SERVICE user, not under the current user.
- The "Delete" button deletes the currently selected license keys.

Display the properties of a license

If a license is selected in the license list, its properties are displayed in the right pane of the window.

Select between different license keys for a single product

More than one license key can be installed for a specific product. The checkbox on the left side in the license list marks the currently active license key.

3.2 Platform considerations

Windows

The license keys are stored in the registry:

- HKLM\Software\PDF Tools AG (for all users)
- HKCU\Software\PDF Tools AG (for the current user)

4 Optimization Process

The main intent of the 3-Heights™ PDF Optimization Tool is to reduce the file size of a PDF document and optimize it for a specific field of application (e.g. Internet, Printing, etc.). For that purpose it offers various options to optimize embedded resources such as fonts or images.

4.1 Images

Relevant Factors for the File Size

The size of an image is basically determined by four factors:

1) The pixel mass: The total amount of pixels the image has. An image with a size of 600 by 800 pixels has 480000 pixels total.
2) The color depth: How many bits are required to describe 1 pixel? The table below gives the answer for different types of images. For example, an RGB image with 600 by 800 pixels requires therefore 600 x 800 x 3 bytes = 1.44 Mbytes in uncompressed format.

Color Space	***Description***	***Bits/pixel***
Bi-tonal	Black and white	1
Indexed	Colors are stored in an index table which usually holds 2 to 256 entries, e.g. GIF.	2-8
Grayscale	Monochrome	8
Color RGB	Color using Red, Green, Blue	24
Color CMYK	Color using Cyan, Magenta, Yellow, Key (=black)	32

3) The compression type: A compression algorithm can compress data (such as an image) to reduce its file size. Such an algorithm belongs to either of the following two classes:

a. Lossless: The original image can be restored exactly.

b. Lossy: The compression modifies the pixels. The original image cannot be restored from the compressed version. This is typically applied to photographic images where the human eye cannot distinguish whether the image was modified. The most common lossy compression is JPEG. The benefit of lossy compression is the higher compression ratio.

See also chapter "Supported image compression types".

4) The content of the image: The simpler the image, the better it compresses. For most compression algorithms a simple image (e.g. completely white) compresses much better than a complex image (e.g. a photo).

Examples

CCITT Fax compression was designed to compress black text written on a white background. The algorithm was optimized under the assumption that a page contains more white pixels than black pixels. Therefore a bi-tonal image with a lot of black does generally not compress as well as in image with more white even if they have the same pixel mass.

JBIG2 compression searches for patterns, and uses them multiple times. For example in a scanned text document the same few dozen of characters are used over and over again. The algorithm is optimized to save frequent patterns more efficiently than rare ones.

Optimizing Raster Images

The 3-Heights™ PDF Optimization Tool offers the following possibilities to optimize images:

1) The pixel mass can be reduced. (It cannot be increased.) This is done by clipping (cropping) the image size to its visible extent and/or by reducing the image resolution.

   The resolution defines how many pixels there are in given length of the image. The most common unit for resolution is dpi: Dots per inch. If an image has a resolution of 200 dpi, it means when displayed at 100% zoom, there are 200 pixels for 1 inch of image. The higher the resolution is, the "sharper" the image. A monitor has usually a resolution of at least 96 dpi, a laser printer of at least 600 dpi. When the file size matters, a common resolution for color and grayscale images in PDF is 150 dpi (usually higher for bi-tonal).

   The process of changing the amount of pixels an image has, is called re-sampling, or down-sampling when the result has less pixels than the original image.

   In the 3-Heights™ PDF Optimization Tool down-sampling is applied by setting a target resolution and a threshold resolution. The default values are 150 dpi for the target resolution and 225 dpi for the threshold resolution. This means every image that has a resolution of 225 dpi or higher is potentially down-sampled to 150 dpi. Of course, the threshold resolution can be set equal to the target resolution. However there are many cases where down-sampling by just a little bit has disadvantages. In particular, lossy images (e.g. JPEG compression)

© PDF Tools AG – Premium PDF Technology

loose visual quality every time they are newly compressed. On top of that the compressed output can be larger than the input because artifacts introduced by the previous compression(s) are now considered as part of the image which needs to be compressed and lead to a worse compression even when the resolution is reduced. Per default, the 3-Heights™ PDF Optimization Tool will, however, prevent such unnecessary re-sampling.

2) The color depth can be modified for color images. The color depth can be left unchanged, set to Grayscale (8 bit), RGB (24 bit) or CMYK (32 bit). It cannot be changed to black and white (1 bit). Note that in certain circumstances, the color depth of the image is not converted, e.g., if the resulting file size increases or if the image is pre-blended with a matte color.

3) The compression can be setup independently for the following three image compression types:

Type	*Description*
Bi-tonal	Black and white images.
Indexed	Images with an indexed (also known as "paletted") color space.
Continuous	Color (RGB and CMYK) images and grayscale images.

Bi-tonal images usually contain text or black and white graphics, indexed images usually contain color graphics such as logos, while continuous images usually contain photographs.

For each of the above image types, several compression algorithms can be set. The 3-Heights™ PDF Optimization Tool tries all the given compression algorithms and takes the one that yields the smallest file size. Note that the more compression algorithms are set, the longer the process of optimizing images will take.

Furthermore, a more conservative image processing strategy can be enabled. This strategy prevents all the compression trials if the image has neither been clipped nor down-sampled nor undergone a color-conversion. Hence, if the image has not been altered, then the original image from the input document is taken.

4) The content of the image cannot be changed directly. However changing the resolution or applying a lossy compression algorithm modifies the content of the image.

Note: Unless forcing of re-compression is enabled, the 3-Heights™ PDF Optimization Tool never increases the file size of an image because it chooses the smallest among all tried compression algorithms and the original image in the input file. This means the 3-Heights™ PDF Optimization Tool cannot be used to "un-compress" embedded images.

Mixed Raster Content (MRC) Optimization for Images

Some raster images – typically scanned documents – consist mainly of text, possibly in several colors and interspersed with some pictures. Such images are difficult to compress with one single compression type because of the diverse or even conflicting features of different parts of the image.

MRC optimization is a way of breaking such images down into parts, such that each part is well suited for one type of a compression algorithm. With this approach, the resulting file size often can be reduced without significantly reducing the visual quality of the document.

Note: MRC optimization can only be enabled for continuous images, i.e. not for bi-tonal images and images with an indexed color space.

MRC optimization may yield unexpected results, e.g. because the input image is not suitable for MRC. As another example, images in the original PDF may be stored as small slices, and MRC optimization fails because the 3-Heights™ PDF Optimization Tool has no option to concatenate such image slices.

A PDF that contains MRC-optimized images is not suited for optical character recognition (OCR) and image extraction.

In the 3-Heights™ PDF Optimization Tool, MRC optimization works in three phases as explained below.

Phase 1: Cutting out Pictures

In this phase, the input image is analyzed and rectangular areas containing photographic features are detected. Each detected region is cut out and placed as a separate image in the resulting PDF.

Depending on the input image it is possible that this phase decides that the whole input image consists of one photographic region covering the whole image. In this case, the second phase (Phase 2: Separation into Layers) is omitted.

On the other hand, it is possible, that actual photographic regions present in the input image are not recognized correctly. This can happen for example if a photographic region contains parts with uniform color.

For the cut-out images, a compression type can be set.

Note: The resulting cut pictures are neither down-sampled nor color-converted.

This first phase is optional and can be switched off.

Phase 2: Separation into Layers

For this second phase the image is not supposed to contain any photographic features. Instead, the image is assumed to consist of text and graphic, potentially with varying color.

Now, the whole image is separated into two layers, a foreground and a background layer. Additionally, a mask is created, which can be thought of as a bi-tonal image that is not displayed directly but tells for each pixel whether to show the foreground layer or the background layer.

Example: Let the image consist of a yellow background with black paragraph text and a title text in red. Then the resulting background layer contains the yellow color only. The foreground layer contains the black text color where the paragraph text is located and the red text color where the title is located. In the mask, pixels for which the foreground layer should be displayed are set to 1, the others are set to 0. I.e. the mask contains 1's where the black and the red text is and 0's everywhere else.

In the resulting PDF the foreground layer, the background layer and the mask are stored as three images and thus are allowed to have different resolution and different

© PDF Tools AG – Premium PDF Technology

compression types. Since all the detailed features have been moved to the mask, it makes sense to down-sample the foreground and background layers and use a low image quality. The mask on the other hand is usually stored with a lossless compression type optimized for text.

Phase 3: Reconstruction

In this phase the results of phase 1 (the cut-out images) and phase 2 (the layers and the mask) are used to synthesize the desired result. If in phase 1, a single photographic region covering the entire image is detected, then the original image is used and the reconstruction is finished. Otherwise, the reconstruction first places the background layer, followed by the foreground layer with the mask. Finally if any cut-images are found they are placed at their respective locations on top of the foreground layer.

4.2 Optimize Fonts

Every text in a PDF document is written with a font. This font can either be embedded or not embedded in the resources of the PDF. Embedded means a font program is embedded that describes how glyphs are drawn. If a font is not embedded the application rendering the PDF (e.g. 3-Heights™ PDF Viewer or Adobe Acrobat) have to select a replacement font. Therefore the visual appearance of text written with an embedded font is determinable, whereas it is not when the font is not embedded.

A font program can be quite large. An embedded font which contains all WinAnsi characters has a size of about 20-100 Kbytes, if it contains a large Unicode range (e.g. Asian Characters) it can be several Mbytes, whereas an non embedded font require much less.

This leads to the following ways to optimize fonts:

1) Remove the embedded Font: Removing embedded fonts can reduce the file size of a document, particularly when the document contains many fonts. Removing fonts is best applied to (PDF-) standard fonts, such as Arial, Courier, Courier New, Helvetica, Times, Times New Roman. Removing fonts should not be applied to barcode fonts or fancy types.

Note: PDF/A requires fonts to be embedded.

2) Subset Fonts: Only keep the information in the font program that is required to render the characters that are actually used in text in this document. All unused characters are removed.

3) Merge Fonts: A document can have the same font, or a subset of it, embedded multiple times. This commonly occurs when multiple input document, are merged into one large output document. The 3-Heights™ Optimization Tool can merge these fonts into one font (if they are mergable).

4.3 Suggested Settings for the Web

When optimizing PDF files for the web, the main goal is to reduce the file size without losing too much visual quality. Additionally files should be linearized, which allows for viewing random pages without download the entire file.

Suggested settings:

```
-c 1 -fb 6,7 -fc 1,8 -oc -od -or -ow -s -m -rs
```

Optionally information can be stripped to further minimize the file size:

```
-sa -si -sm -ss -ss -st -sw
```

If encrypting:

```
-o <ownerpassword> -p pf
```

4.4 Suggested Settings for Printing

Suggested settings:

```
-c 2 -dt -1 -od -or -s -m
```

Optionally, information can be stripped to further minimize the file size:

```
-sa -sf -sm -ss -ss -st -sw
```

If encrypting:

```
-o <ownerpassword> -p pd
```

© PDF Tools AG – Premium PDF Technology

5 Reference Manual

5.1 Service Control Commands

These options are used to control the service. The *create* and *delete* functions require administrator rights. The *start* and *stop* functions require operator rights.

-c Create Service

The optimization service is created using the option –c. The default user of the service is the LocalSystem. The user can be changed as long as the service is not started.

```
C:\pdf-tools\bin\pdfoptimizesvr –c
```

Important: It is essential that *pdfoptimizesvr.exe* be on a **non-mapped** drive.

-s Start Service

Once created, the optimization service can be started with the option –s.

```
pdfoptimizesvr –s
```

-a Pause Service

This option pauses the service.

```
pdfoptimizesvr –a
```

-o Continue Service

This option resumes the service.

```
pdfoptimizesvr –o
```

-t Stop Service

To stop the service, use the option –t.

```
pdfoptimizesvr –t
```

If "stop" if called while the service is "running", the current job (all pages) will be finished, after that the service is stopped.

If the service was "paused" before calling "stop", the current page while be finished processing. After that page, the job is aborted.

© PDF Tools AG – Premium PDF Technology

-d Delete Service

The optimization service can be deleted with the option –d.

```
pdfoptimizesvr –d
```

-q Query Current Status of Service

This option returns the current status of the service.

```
pdfoptimizesvr -q
The service starts automatically during system startup.
The service is stopped.
[PdfOptimizeSvr] QueryService: The operation completed successfully.
```

-i List Options for the Configuration File

This options lists all available options to configure the service.

```
pdfoptimizesvr -i
```

-x Run as Executable

With this option, the optimization service runs as an executable instead of as a Windows Service. It provides the same functionality as long as the exe is running.

```
pdfoptimizesvr –x
```

© PDF Tools AG – Premium PDF Technology

5.2 File and Folder Options

Configuration File *PdfOptimizeSvr.ini*

The configuration file defines the setting for the watched folders. It is read upon starting the service.

`[PdfOptimizeSvr]`	required
`Autodelete=...`	optional, `true` or `false`
`AutodeleteAll=...`	optional, `true` or `false`
`LogPath=...`	optional, if used must be a path like `C:\mypath\log` or the keyword `EventLog`
`PollingInterval=...`	optional, value in milliseconds, default `1000`
`JobPrefix=...`	optional, `true` or `false`
`Threads=n`	required
`Thread1=-w ...`	required
`Thread2=-w ...`	
`...`	
`Threadn=...`	There must be exact as many threads as defined in Threads=n.

Example:

```
[PdfOptimizeSvr]
Autodelete=true
LogPath=EventLog
JobPrefix=false
Threads=2
Thread1=-w C:\WebOptimize -c 1 -dt 120 -dr 96 -q 75 –od -or -ow –ss -st
Thread2=-w C:\PrintOptimize -dt 1500 -dr 1200 –od -or
```

-lk Set License Key

Pass a license key to the application at runtime instead of installing it on the system.

-w Set the Watched Folder

Use the option –w to define the path of the watched folder. This path should not contain mapped drives, since other users (such as LocalSystem) do not recognize them. This parameter must always be the first parameter of a thread.

```
-w C:\PdfOptimizeSvr\WebOptimize
```

Note that the service supports path lengths including file name of up to 258 characters. This includes the 21 characters of the job ticket. If a file name exceeds this value, its file name is truncated at the end of the file name and before the file extension. It is therefore suggested that watched folder names are kept reasonably short.

-wd Set the Drop-In Folder

By default the drop-in folder is equal to the folder defined as watched folder using the option –w. If the input files should be taken from a different folder, this can be configured using –wd. All folders created by service including the output folder are at the directory defined by –w.

```
-wd C:\SomePath\DropIn
```

-wfs Process only files with certain extensions

By default, the service tries to process all files dropped into the drop-in folder, regardless of the extension. With this option, the processing can be restricted to a set of known file extension.

Example: *Restrict the processing to JPEG and PNG files.*

```
-wfs .jpeg.jpg.png
```

-wfi Ignore files with certain extensions

By default, the service tries to process all files dropped into the drop-in folder, regardless of the extension. With this option, files with certain file extensions ca be ignored.

Example: *Ignore temporary files.*

```
-wfi .temp.tmp
```

-p Set the Permission Flags

This option sets the permission flags. It is only usable in combination with encrypted documents. By default no permissions are granted. The permissions that can be granted are listed in Table: Permission Flags.

Table: Permission Flags

Parameter	*Description*
`p`	allow printing (low resolution)

© PDF Tools AG – Premium PDF Technology

m		allow changing the document
c		allow content copying or extraction
o		allow commenting
f		allow filling of form fields
s		allow content extraction for accessibility
a		allow document assembly
d		allow high quality printing
-1	default	allow everything (all permissions are granted)
0		allow nothing (no permissions are granted)

The following command sets the owner password to "owner" and the permission flags to allow "printing in low resolution" and "allow form filling".

```
-o owner -p pf
```

Note that "high quality printing" requires the "low resolution printing" flag to be set as well:

```
-o owner -p pd
```

For further information about the permission flags, see PDF Reference Manual section 3.5.2.

-pw Read an Encrypted PDF File

When the input PDF document is encrypted and has a user password set, (the password to open the PDF) either the user or the owner password must be provided to open the document.

If for example the user password is "userpwd", then the command would look like this:

```
-pw userpwd
```

When a PDF is encrypted with a user password, and no password is not provided or it is incorrect, the document cannot be process. Instead it will generate an error message in the log file and move the file to the folder "Failed".

-o Set the Owner Password

The owner password is required to change the security settings of the document.

```
-o ownerpassword
```

-u Set the User Password

The user password is required to open the document.

```
-u userpassword
```

5.3 Optimization Options

Compression Values for Images

The compression value takes a number from 0 to 8. The supported compression algorithms are listed below:

0 No Compression (Raw)

The raw format results in an uncompressed image. There is no real benefit using Raw.

Compression: None
Color depth: any

1 DCT (JPEG) Compression

The DCT (Discrete Cosine Transformation) is commonly used for image processing, especially for lossy data compression.

Compression: High, Lossy
Color depth: 8, 24
Application area: Color images

2 Flate (ZIP) Compression

DEFLATE is a lossless data compression algorithm that uses a combination of the LZ77 algorithm and Huffman coding.

Compression: High, Lossless
Color depth: 8, 24
Application area: Images

3 LZW (Lempel-Ziff-Welch) Compression

LZW (Lempel-Ziv-Welch) is an implementation of a lossless data compression algorithm created by Abraham Lempel and Jacob Ziv. It was published by Terry Welch in 1984 as an improved version of the LZ78 dictionary coding algorithm developed by Lempel and Ziv. There are certain countries where this algorithm is still protected by a copyright.

Compression: High, Lossless
Color depth: 2-8
Application area: Grey-scale images, artificial images

4 CCITT Fax Group 3 Compression

1-dimensional version of the CCITT Group 3 Huffman encoding algorithm.

Compression: Low, Lossless
Color depth: 1
Application area: Line-art image, bi-tonal, faxes

5 CCITT Fax Group 3 2D Compression

2-dimensional version of the CCITT Group 3 Huffman encoding algorithm.

© PDF Tools AG – Premium PDF Technology

Compression: Medium, Lossless
Color depth: 1
Application area: Line-art image, bi-tonal, faxes

6 CCITT Fax Group 4 Compression

An advanced version of a bi-tonal algorithm based on the CCITT Fax Group 3 2D compression.

Compression: Medium, Lossless
Color depth: 1
Application area: Line-art image, bi-tonal, faxes

7 JBIG2 Compression

JBIG2 is an image compression standard for bi-level images, developed by the Joint Bi-level Image Experts Group. It is suitable for both lossless and lossy compression.

Compression: High, Lossless (Q=100)/Lossy
Color depth: 1
Application area: Line-art image, bi-tonal

8 JPEG2000 Compression

JPEG 2000 is a wavelet-based image compression standard. It was created by the Joint Photographic Experts Group committee with the intention of superseding their original discrete cosine transform-based JPEG standard.

Compression: High, Lossless (Q=100)/Lossy
Color depth: 8, 24
Application area: Images

9 MRC Optimization

In contrast to the values 0-7, this is not a single compression format. Instead, this enables MRC optimization on color and monochrome images. (See Mixed Raster Content (MRC) Optimization for Images.)

Application area: Scanned documents

10 Compression of the Source Image

In contrast to the values 0-7, this is not a single compression format. Instead, this tells 3-Heights™ PDF Optimization Service to use the same compression as the original input image.

-c Set the Color Conversion

This switch allows for converting raster images from one color space into another. E.g. it allows for converting all RGB images to CMYK images.

This switch does not have any impact on objects other than raster images that use color spaces, such as vector graphics or text. Color key masked images are not color converted. Pre-blended images can be converted from RGB to Grayscale, if the force conversion feature is set.

© PDF Tools AG – Premium PDF Technology

Use the switch `-c` followed by one of the parameters in the table listed below:

Table: Color Conversion

Parameter	*Conversion*	*Color values*
0 default	Don't convert colors	
1	Convert to ICE sRGB colors	red, green, blue
2	Convert to CYMK color (using profiles)	cyan, yellow, magenta, key
3	Convert color images to grey scale	grey

Example: *To convert all embedded color images that use the RGB color space to images of the CMYK color space, use the following command:*

```
-c 2
```

-cff Compress Type1 fonts (convert to CFF)

Convert embedded Type1 (PostScript) fonts to Type1C (Compact Font Format). This reduces the file size.

-cms Set the color management engine

Set the Color Management System (CMS) Engine used for color conversion (option –c). Supported engines are:

- *"None":* No CMS is applied. This results in the maximum possible contrast.
- *"Neugebauer":* The Neugebauer algorithm efficiently converts CMYK to RGB. It does not need any color profiles. The results however look similar to conversion using color profiles.
- *"MSICM":* The Microsoft ICM Engine.
- *"lcms" (default):* Use ICC color profiles (default profiles are used for all unmanaged device color spaces).
- *FileName*: Providing a file name, a configurable version of the Neugebauer algorithm is applied. The coefficients can be defined in the text file. The Neugebauer algorithm mixes the colors based on the amount of color and the corresponding weighted coefficient. Altering the values for a pure color specifically changes the result for this pure color. The color transition remains smoothly.

Set the Resolution and Threshold per Image Type

The target resolution values can be set individually for different types of images using the following switches followed by a numerical parameter (default: 150):

-dbr DPI for bi-tonal images
-dcr DPI for color images
-dmr DPI for monochrome images

The threshold values can be set with these switches followed by a numerical parameter (default 225):

-dbt DPI for bi-tonal images
-dct DPI for color images
-dmt DPI for monochrome images

-dr Set the Resolution in DPI

Set the target resolution after re-sampling in dots per inch (dpi). Only those images with a resolution value higher than the threshold value, which is set with option `-dt`, will be processed. The default target resolution is 150 dpi. Pre-blended images, images with a color key mask, masks, and soft mask images are not re-sampled.

Example: *In order to down-sample all raster images with a resolution greater than 150 dpi to 75 dpi, apply the following:*

```
-dt 150 -dr 75
```

-dt Set the Threshold in DPI

This switch defines the minimum resolution an image must have to be optimized. The threshold value for re-sampling raster images is used in conjunction with the switch `-dr`, which sets the actual target resolution for those re-sampled images.

The threshold resolution must be equal or higher than the target resolution. If the value is set to –1, re-sampling is turned off. The default threshold resolution is 225 dpi.

Example*: Down-sample all raster images with an original resolution higher or equal to 150 dpi to a new resolution of 75 dpi:*

```
–dt 150 –dr 75
```

Example: *To disable re-sampling, set the threshold value, set it to –1.*

```
–dt –1
```

If the size (in terms of bytes) of the re-sampled image is larger than its original size, the original image is kept instead.

-fb Set the Compression Types for Bi-tonal Images

This option affects only bi-tonal (black and white) images. The switch `-fb` is followed by a comma-separated list of numerical values (no spaces allowed in the list). The following values are possible:

Table: Bi-tonal Compression Types

Parameter	*Compression Filter*
0	RAW data
2	Flate (ZIP) compression

© PDF Tools AG – Premium PDF Technology

3		Lempel-Ziv-Welch (LZW) compression
4		CCITT Fax Group 3 compression
5		CCITT Fax Group 3 2D compression
6	default	CCITT Fax Group 4 compression
7		JBIG2 compression
10		Take the compression type from the original image in the input PDF

Example: *To let the 3-Heights™ PDF Optimization Service try CCITT Group 3 compression, JBIG2 compression and the compression that is used in the source image of the original file use the following command:*

```
-fb 3,7,10
```

The above command makes the 3-Heights™ PDF Optimization Service go through all bi-tonal images and processes each image individually as follows. All the given compression algorithms are executed. If the input image has a compression different from CCITT Fax Group 3 and JBIG2, then the compression of the input image is also executed. As a result, several candidate versions are obtained. Now a choice is made among all these versions and the original image based on the size in bytes. The smallest candidate is chosen and used in the output document.

-fc Set the Compression Types for Color and Grayscale Images

This option affects normal color images (RGB and CMYK) as well as grayscale (monochrome) images. The switch `-fc` is followed by a comma-separated list of numerical values (no spaces allowed in the list). The following values are possible:

Table: Color / Monochrome Compression Types

Parameter		*Compression Filter*
0		RAW data
1	default	DCT (JPEG) compression
2		Flate (ZIP) compression
8		JPEG2000 compression
9		Perform MRC optimization (See Mixed Raster Content (MRC) Optimization for Images)
10		Take the compression type from the original image in the input PDF

Example*: **:** To let the 3-Heights™ PDF Optimization Shell try JPEG compression, JPEG2000 compression and the compression that is used in the source image of the original file use the following command:*

```
-fc 1,8,10
```

The above command makes the 3-Heights™ PDF Optimization Service go through all color and grayscale images and processes each image individually as follows. All the given compression algorithms are executed. If the input image has a compression

different from JPEG and JPEG2000, then the compression of the input image is also executed. As a result, several candidate versions are obtained. Now a choice is made among all these versions and the original image based on the size in bytes. The smallest candidate is chosen and used in the output document.

***Example**: The following command performs MRC optimization for color and grayscale images:*

```
-fc 9
```

If the result takes more memory than the original image in the input document, then the original image is used. See also Mixed Raster Content (MRC) Optimization for Images.

-ff Force Compression Conversion

If set, images are always recompressed. If not set (default), images are only recompressed if the resulting image is smaller than the original, i.e. occupies less bytes to store in the file.

-fi Set the Compression Types for Indexed (Paletted) Images

This affects only images with an indexed color space. This type of color space is sometimes used for color graphics and logos. The switch `-fc` is followed by a comma-separated list of numerical values (no spaces allowed in the list). The following values are possible:

Table: Indexed Compression Types

Parameter		*Compression Filter*
0		RAW data
2	default	Flate (ZIP) compression
3		Lempel-Ziv-Welch (LZW) compression
10		Take the compression type from the original image in the input PDF

***Example**: **:** To let the 3-Heights™ PDF Optimization Service try Flate compression and LZW compression use the following command:*

```
-fi 2,3
```

The above command makes the 3-Heights™ PDF Optimization Service go through all images with indexed color space and processes each image individually as follows. All the given compression algorithms are executed. As a result, two candidate versions are obtained. Now a choice is made among these two versions and the original image based on the size in bytes. The smallest candidate is chosen and used in the output document.

-fv Set the Minimum PDF Version

This option allows to set the minimum PDF version of the created PDF output file. Supported values are 1.1 to 1.7. (PDF 1.4 corresponds to Acrobat 5, PDF 1.5 to Acrobat 6, etc.) There are three parameters that influence the version of the PDF output file:

The value set using the option –fv

The PDF version of the input file

Other settings in the optimization (JBIG2 requires PDF 1.4, JPEG2000 requires PDF 1.5)

The maximum of the three values above sets the PDF version in the output file.

Examples:

1. Input PDF is version 1.5 and the following setting is used

   ```
   -fv 1.4
   ```

   The output file is PDF version 1.5.

2. Input PDF is version 1.4 or lower and the following setting is used:

   ```
   -fv 1.4
   ```

   The output file is PDF version 1.4.

3. Input PDF is version 1.3 and the following setting is used:

   ```
   -fv 1.4 -fc 8
   ```

   If the input contains color images to which JPEG2000 compression is applied, the output file will be version 1.5. Otherwise it will be version 1.4.

-id Set Value in the Document Information Dictionary

Set the value of an info entry key. Examples for keys are "Author", "Subject", "Title", "Producer" or custom attributes.

Example: *Set the title:*

```
-id Title "My Title"
```

-isc Enable Conservative Image Processing Strategy

If set, then the different compression types given with -fb, -fc, and -fi are omitted if the current image is neither clipped nor color-converted nor down-sampled. Thus, the original image in the input document is conserved if it has not been altered.

-m Merge Embedded Font Programs

Font programs can be merged, if they originate from the same font, e.g. they are of the same type, have the same name and encoding. Merging of Type1 (PostScript) and TrueType fonts is supported.

-ml Set Compression Type for MRC Layers

This option affects only MRC foreground and background layers. The option is followed by a single numerical value indicating the compression type to use for MRC foreground and background layers. For possible values see "Table: Color / Monochrome Compression Types" in the description of -fc. The default is 1 (JPEG compression).

See also Mixed Raster Content (MRC) Optimization for Images.

-mlr Set the Resolution in DPI for MRC Layers

This option affects only MRC foreground and background layers. The option is followed by a numerical value that indicates the target resolution in DPI of MRC layers after down-sampling. The default is 70.

See also Mixed Raster Content (MRC) Optimization for Images.

-mlq Set the Image Quality for MRC Layers

This option affects only MRC foreground and background layers. The option is followed by a numerical value between 0 and 100 to be used as the image quality for MRC foreground and background layers when using a lossy compression for these layers. The default is 10.

See also Mixed Raster Content (MRC) Optimization for Images.

-mm Set Compression Type for the MRC Mask

This option affects only MRC masks. The option is followed by a single numerical value indicating the compression type to use for MRC masks. For possible values see "Table: Bi-tonal Compression Types" in the description of -fb. The default is 6 (CCITT Fax Group 4 compression).

See also Mixed Raster Content (MRC) Optimization for Images.

-mp Set Compression Type for MRC Cut-Out Pictures

This option affects only cut-out images when doing MRC optimization. The option is followed by a single numerical value indicating the compression type to use for MRC cut-out pictures. For possible values see "Table: Color / Monochrome Compression Types" in the description of -fc. The default is 1 (JPEG compression).

See also Mixed Raster Content (MRC) Optimization for Images.

-o Set the Owner Password

The owner password is required to change the security settings of the document. In order to apply permission flags, an owner password must be set. Permission flags are set with the switch **-p**.

Example: *Encrypt a document and set the owner password to "owner".*

```
-o mypassword
```

-oc Clip Images

Images in PDF documents can be clipped. This means that only part of the image is visible, whilst the rest is hidden. The switch **-oc** detects these images, reduces their size the area that is actually displayed and replaces the original image by the reduced image. Pre-blended images are not clipped.

Setting **-oc** activates the **-od** option.

-od Optimize Resources

Optimize the resources of the PDF, such as images, color spaces, or fonts. If set, unused resources are removed. Also content streams are re-built.

-ol Linearize Only

Do not apply any optimizations, but linearize the file. This can be significantly faster than the option **-ow**. See -ow for more information.

-or Remove Redundant Objects

This option removes redundant objects. I.e. it identifies duplicates of objects and merges them.

-ow Linearize the Output File

Add so called linearization tags to the document. A linearized document has a slightly larger file size than a non-linearized file, and provides the following features (among others):

- When a document is opened through a PDF viewing application plug-in for an Internet browser, the first page can be viewed without downloading the entire PDF file.
- When another page is requested by the user, that page is displayed as quickly as possible and incrementally as data arrives, without downloading the entire PDF file.

Note: In order to make use of a linearized PDF file, the PDF must reside as a 'file' on the web-server. It must not be streamed.

-p Set the Permission Flags

This option sets the permission flags. It is only usable in combination with encrypted documents, i.e. an owner password must be set. By default all permissions are granted. The permissions that can be granted are listed in the table below.

Table: Permission Flags

Parameter	*Description*
p	allow printing (low resolution)
m	allow changing the document

`c`		allow content copying or extraction
`o`		allow commenting
`f`		allow filling of form fields
`s`		allow content extraction for accessibility
`a`		allow document assembly
`d`		allow high quality printing
`-1`	default	allow everything (all permissions are granted)
`0`		allow nothing (no permissions are granted)

The parameter 0 cannot be combined with other flags. The parameter –1 is the default, it cannot be set explicitly. In order to combine multiple permissions concatenate them to one string.

Example: *The following sets the owner password to "owner" and the permission flags to allow "printing in low resolution" and "allow form filling".*

```
-o owner -p pf
```

Example: *"High quality printing" requires the standard printing flag to be set too.*

```
-o owner -p pd
```

For further information about the permission flags, see PDF Reference Manual section 3.5.2.

-pw Read an Encrypted PDF File

When the input PDF file is encrypted and has a user password set, (the password to open the PDF) the password can be provided as parameter of the switch `-pw`.

Example: *The input PDF document is encrypted with a user password. Either the user or the owner password of the input PDF is "mypassword". The command to process such an encrypted file is:*

```
-pw mypassword
```

When a PDF is encrypted with a user password and the password is not provided or is incorrect, the 3-Heights™ PDF Optimization Shell cannot read and process the file. Instead it will generate the following error message in the log file:

```
Password wasn't correct.
```

-q Set the Compression Quality

Set the compression quality index for lossy compression methods. This option only applies to JPEG, JPEG2000 and JBIG2 images. A lower value results in a smaller file size but the images are of poorer visual quality. A higher value results in better visual quality, but also a larger file size.

The supported values range from 1 (lowest) to 100 (highest). The default is 75. For images compressions that support lossless compression (JPEG2000 and JBIG), a value

of 100 corresponds to lossless compression, any other value represents lossy compression. JBIG2 only supports values that are multiples of 10 (10, 20, … 100).

Example: *The following command sets the quality index to 50. All images types which support the quality parameter are recompressed with this quality index.*

```
-q 50
```

-rs Remove Embedded Standard Fonts

This option removes all embedded standard fonts and replaces them with one of the 14 PDF Standard Fonts. The following font families are removed:

Arial	CourierNewPS	Times	ZapfDingbats
Courier	Helvetica	TimesNewRoman	
CourierNew	Symbol	TimesNewRomanPS	

and their derivatives (they are different for different font families) such as:

Arial,Bold	Arial-Bold	Arial-Italic	ArialMT
Arial,BoldItalic	Arial-BoldItalic	Arial-BoldMT	Courier-Bold
Arial,Italic	Arial-BoldItalicMT	Arial-ItalicMT	Courier-Oblique

A PDF Viewer must be able to display standard fonts correctly, even if they are not embedded. Therefore using this option should not visually alter the PDF when it is displayed. Un-embedding a font decreases the file size.

Un-embedding the font works based on the font's Unicode information. I.e. the un-embedded font's characters are mapped to those of the original font with the same Unicode. Therefore, only fonts with Unicode information will be un-embedded by the 3-Heights™ PDF Optimizer. However, if a font's Unicode information is not correct, un-embedding may lead to visual differences. Whether or not a font's Unicode information is correct can be verified by extracting text that uses the font. Suitable tools for this purpose are for instance the 3-Heights™ PDF Extract Tool or an interactive PDF viewer. If the extracted text is meaningful, the font's Unicode information is correct and un-embedding of the font will not lead to visual differences.

-s Subset Fonts

Embedded fonts can be subsetted. Subsetting refers to only storing those character glyphs of the font that are actually used. Unused character glyphs are removed. The advantage is that the file size can be reduced this way (in particular for Asian fonts). The downside is that if text is to be edited, only the characters of the subsetted font can be used.

Strip the File

Remove parts of the PDF file. The following parts of a PDF can be stripped:

-sa Strip article threads.

-sf Strip and flatten form fields and annotations. (This removes interactive features of the PDF.)

-si Strip alternate images (variant representations of the base image).

-sm Strip meta data.

-sp Strip page piece info (private application data).

-ss Strip document structure tree (incl. markup).

-st Strip embedded thumbnails.

-sw Strip spider (web capture) info.

-se Strip everything (all of the above).

-u Set User Password

Set the user password of the document. If a document which has a user password is opened for any purpose (such as viewing, printing, editing), either the user or the owner password must be provided.
Someone who knows the user password is able to open and read the document. Someone who knows the owner password is able to open, read and modify (e.g. change passwords) the document. A PDF document can have none, either, or both passwords.

Example: *Encrypt a document with a user and an owner password.*

```
-u userpassword –o ownerpassword
```

5.4 Service Settings

Autodelete Successfully Processed Files

When a job has successfully been processed, a new PDF document is created in the folder *Optimized* and the original PDF document is by default deleted. In order to retain the original PDF document after successful processing, set the value of "Autodelete" in the configuration file to "False". The original PDF document will then be moved from the folder *Jobs* to the folder *Succeeded*. Failed documents are still moved to the folder *Failed*.

```
[PdfOptimizeSvr]

AutoDelete=False
```

To delete failed documents as well use the following setting:

```
[PdfOptimizeSvr]

AutoDeleteAll=True
```

Job Number Prefix

At the time when a document is copied from the watched folder to the *Jobs/* sub folder, it is renamed and added a 21 character prefix containing a time stamp: Job-{8 digits}-{8 digits}_ Something like: "Job-01C61DD4-E72E1BCE_". The job number prefix ensures the same document (same name) can be processed multiple times.

Adding the prefix can be prevented with the following line in the configuration file:

```
[PdfOptimizeSvr]
JobPrefix=False
```

Logpath

Log-messages created by the service are by default written to the sub-directory *log*. To alter the directory, add a line similar as shown below to the configuration file:

```
[PdfOptimizeSvr]
LogPath=C:\path\log
```

Messages created by the service can be added to the system's application event log instead of written to a log file. This is achieved by adding the following line to the configuration file:

```
[PdfOptimizeSvr]
LogPath=EventLog
```

The system's application log event will then log messages similar as shown below:

- CreateService: The operation completed successfully.
- StartService: The operation completed successfully.

Note that the messages are only fully accessible while the service is created. Otherwise a message as shown below is displayed:

- The description for Event ID (1) in Source (PdfOptimizeSvr) cannot be found. The local computer may not have the necessary registry information or message DLL files to display messages from a remote computer. The following information is part of the event: DeleteService: The operation completed successfully.

Polling Interval

The polling interval defines the time in milliseconds that the polling-thread pauses between two polls. The time passing until the same watched folder is polled again (maximum pick-up time) is: The value of PollingInterval plus the actual time it takes to poll all watched folders. The higher the polling interval, the lower the network traffic, and the longer it takes until documents are picked up.

Suggested values for the polling intervals are 1000 to 10000 milliseconds.

```
[PdfOptimizeSvr]
PollingInterval=5000
```

6 Tips, Tricks and Troubleshooting

6.1 The Output File Is Still Too Large

First and foremost it is important to understand what kind of content there is in the document. There is no point in trying to optimizing fonts when the document contains scanned images only. Document properties, such as embedded fonts and images can be listed using the corresponding listing functions (-li, -lf).

General optimization:

Remove redundant objects and optimize resources using –rs –od can always be set.

For images:

1. Try setting a lower threshold and a lower dpi for the images, e.g. rescale all images with a dpi greater than 72 dpi to 50 dpi:

   ```
   -dt 72 -dr 50
   ```

2. You could also try reducing the quality of the jpeg images with the quality option -q. In many cases using a lossy compression is not significant for viewing:

   ```
   -q 60
   ```

3. Verify what image compression algorithms are applied, the smallest file sizes are usually achieved using JPEG or JPX for grey-scale and color images and JBIG2 for bi-tonal images. When using JPEG, the quality should be at least 75, when using JPX, it can be set as low as 25.

   ```
   -fc 8 –fm 8 –fb 7 –q 30
   ```

For fonts:

4. Apply subsetting to fonts. This means all glyphs of characters that are unused are removed from the font.

   ```
   –s
   ```

5. Remove non-symbolic embedded fonts. Keep in mind that the appearance when rendering a PDF document with non-embedded non-PDF Standard Fonts is unpredictable.

   ```
   -rs –s -m
   ```

6.2 The Output File Is Larger Than the Input File

1. The 3-Heights™ PDF Optimization Service also repairs corrupt documents to a certain extent. This means if relevant data is missing it is recovered. This could possibly lead to a larger file size.
2. If linearization is applied, there is information added to the document. This information contains hints for the browser plug-in, and allows it to specifically download only those objects relevant for displaying a certain page. The linearization information can increase the file size by about 1 to 10%.

© PDF Tools AG – Premium PDF Technology

6.3 The Selected Compression Type is Not Applied

1. Not all compression types can be applied to all types of images. Check the tables "Table: Bi-tonal Compression Types", "Table: Color / Monochrome Compression Types", and "Table: Indexed Compression Types".
2. The optimization is only applied if it reduces the files size, therefore an image cannot be re-compressed with a new compression that uses more disc space than the original compression. This behavior can, however, be switched off with -ff.

6.4 The Output Document Is Not Encrypted

In order to encrypt the output document, set an owner password using the switch **-o** and permission flags using the switch **-p**.

Example to set the owner password to "mypassword" and do not grant any permissions:

```
-o mypassword -p 0
```

It is not possible to inherit the owner or user password or the permission flags from the input document.

© PDF Tools AG – Premium PDF Technology